



第8届 AI+ 研发数字峰会

拥抱 AI 重塑研发 AI+ Development
Digital Summit

11月14-15日 | 深圳





AI+ PRODUCT INNOVATION SUMMIT 01.16-17 · ShangHai

2026 AI+产品创新峰会



扫码查看会议详情



Track 1: AI 产品战略与创新设计

从0到1的AI原生产品构建

论坛1: AI时代的用户洞察与需求发现

论坛2: AI原生产品战略与商业模式重构

论坛3: Agentic AI产品创新与交互设计



Track 2: AI 产品开发与工程实践

从1到10的工程化落地实践

论坛1: 面向Agent智能体的产品开发

论坛2: 具身智能与AI硬件产品

论坛3: AI产品出海与本地化开发



Track 3: AI 产品运营与智能演化

从10到100的AI产品运营

论坛1: AI赋能产品运营与增长黑客

论坛2: AI产品的数据飞轮与智能演化

论坛3: 行业爆款AI产品案例拆解

2-hour Speech: 回归本质



用户洞察的第一性

——2小时思维与方法论工作坊

在数字爆炸、AI快速发展的时代，
仍然考验“看见”的“同理心”

Panel 1: 出海前瞻



“出海避坑地图”圆桌对话

——不止于翻译：

AI时代的出海新范式

Panel 2: 失败复盘



为什么很多AI产品“叫好不叫座”？

——从伪需求到真价值：

AI产品商业化落地的关键挑战

智能重构产品 数据驱动增长
Reinventing Products with Intelligence, Driven by Data

80+
业界大咖
汇聚产品大咖的真知灼见

40+
创新案例
开拓全新AI+产品视角

10+
分论坛
涵盖产品到商业增长的全链路

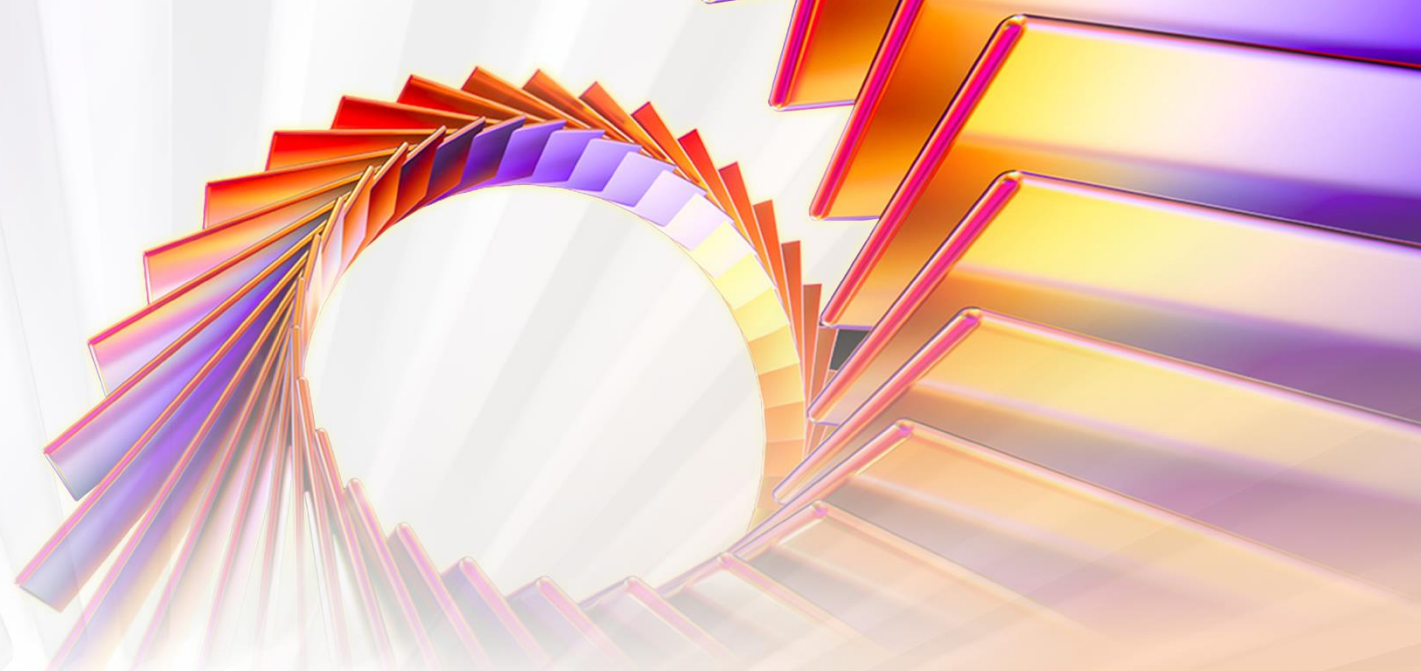
800+
听众规模
共同探索产品的智能重构



| 11月14-15日 | 深圳

第8届 AI+研发数字峰会

拥抱 AI 重塑研发 AI+ Development
Digital Summit



Qoder Coding Agent 探索上下文工程的能力边界

陈鑫 | 阿里巴巴资深技术专家



陈鑫

阿里巴巴资深技术专家

花名神秀，致力于企业研发效率、产品质量、DevOps、AI Coding 方向研究和探索。曾担任过阿里集团研发工具平台Aone负责人、阿里云云效负责人、通义灵码负责人。带领团队不断挑战研发效能提升上限，打造世界领先的开发者工具产品，普惠万千中国软件企业。

目录 CONTENTS

- I. AI Coding 智能体技术发展趋势
- II. 真实软件开发过程中的痛点与思考
- III. Qoder 上下文工程技术优化实践
- IV. 未来技术挑战与应对

PART 01

AI Coding 智能体技术发展趋势

▶ AI Coding 已经全面进入 Agent 时代

模型能力发展为产品带来更多可能性

Qwen3 Max

1

TRILLION
PARAMETER AI MODEL

Qoder Agent VS Code Completion

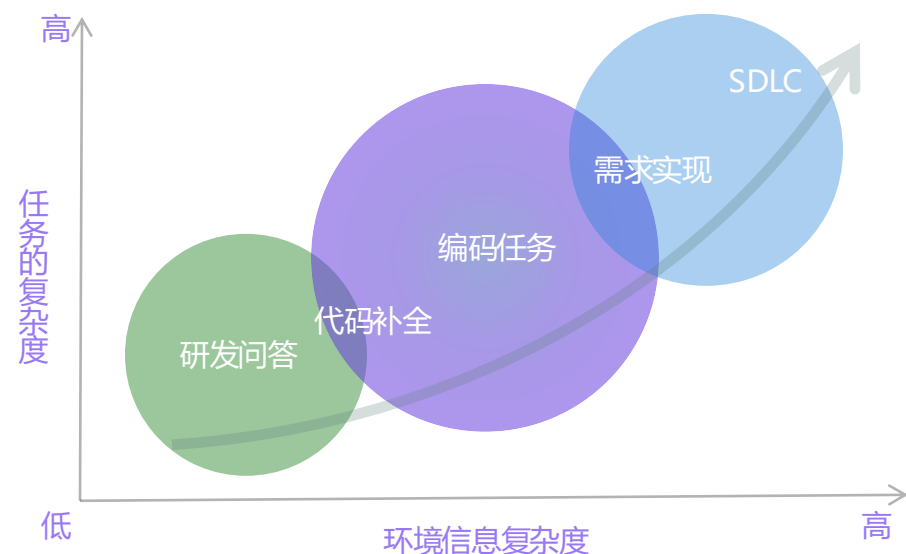
100 vs 1

Lines Accepted



► Coding Agent 正在改变传统软件研发流程

应用场景在扩大，产品边界在延伸



Qoder IDE x Qoder Agent

围绕同一个 Repo 进行任务协同。本地是以人为主体的决策修改，远端是以 Agent 为主体的完成研发任务

DevOps x Qoder Agent

基于非交互式 CLI、SDK 方式，传统 DevOps 工具与 Coding Agent 进行集成，增强现有软件工作流，从流程人主驱动进化到 AI 驱动。

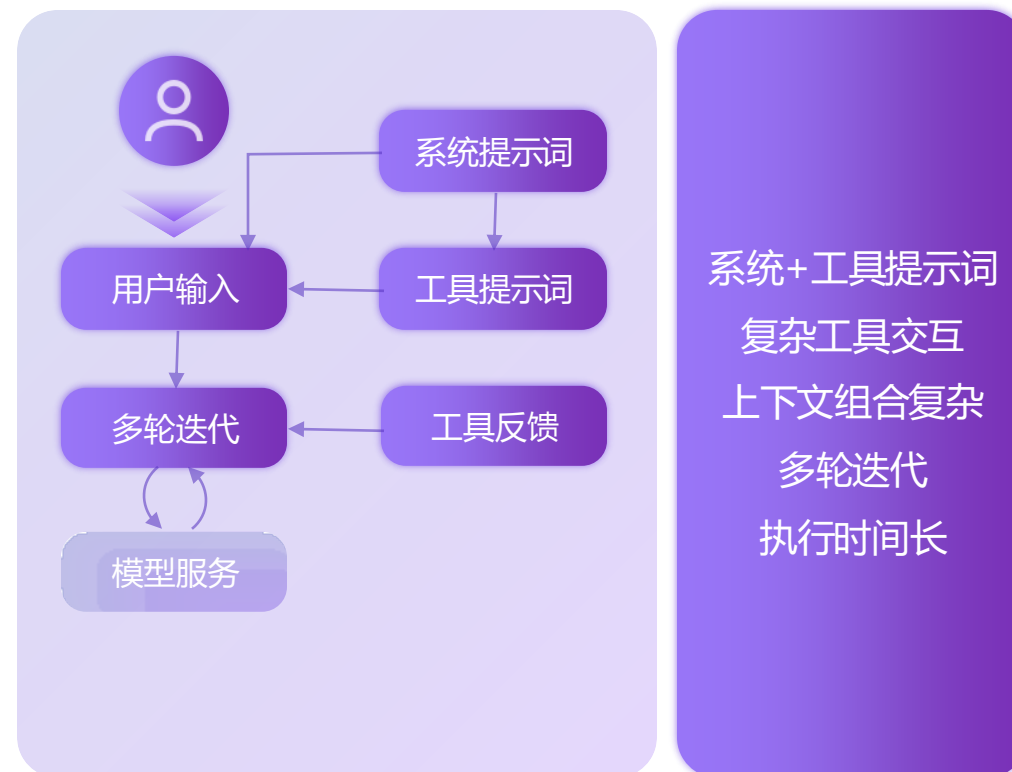


▶ 上下文工程成为 Agent 效果的核心

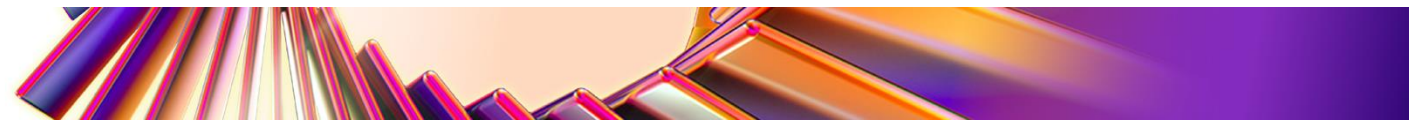
提示词工程



上下文工程



上下文成为 Agent 效果优的核心



PART 02

真实软件开发过程中的痛点与思考

► 真实软件开发过程中的痛点

Top 20 AI Prompt Programming Languages

排名	趋势	提示词语言	市场占有率
1	—	给我生成完整可运行的代码	18.762%
2	↑	用中文回答	16.543%
3	↓	我说了要复用现有接口	14.891%
4	↑↑	别又给我生成一堆TODO	12.937%
5	↓	为什么又引入新依赖	10.984%
6	—	这个错误你上次就犯过	9.215%
7	↑	别自作聪明优化	7.638%
8	↓↓	这个API根本不存在	6.492%
9	↑	还是报错	5.871%
10	↑	注释和代码对不上	4.923%
11	↓	就改这里别的别动	4.156%
12	↑↑↑	不要用any啊大哥	3.784%
13	—	能先跑起来再说吗	3.291%
14	↑	你这代码根本编译不过	2.847%
15	↓	按照我给的例子写	2.563%
16	NEW	求你别改我的命名规范	2.194%
17	↓↓	这不是我要的功能	1.928%
18	↑	只要核心逻辑就行	1.675%
19	↓	别一直import你编的库	1.432%
20	NEW	能不能有点记忆力	1.208%

开发者的期待

生成可编译的准确代码

遵循我的个人/工程习惯

复用现有的代码资产

不犯同样的错误

不自作主张积极过度

Coding Agent 存在的问题

大模型能力不足

工程知识信息不足

无法记忆偏好和教训

没有编程规范指导

开发者没有清晰表达诉求

- 大量的暗知识，知识不对齐：人与人知识不对齐，导致协作成本高；人与 AI 知识不对齐，导致 AI 效果不达预期
- 大量重复、繁琐的 Coding 工作占据了开发者大部分时间，反而在需求澄清意图识别上投入精力过少，导致返工率高
- 与 AI 协同，多采用对话式沟通，过程中不断校正完成代码编写，这种模式下 AI 的工作时间上限是人类的工作时间

Knowledge Visibility

隐性知识显性化，促进真实软件工程可见性提升，依赖代码和提交历史还原知识，降低人的认知负载，完成人与 AI 的知识对齐

Spec-Driven Development

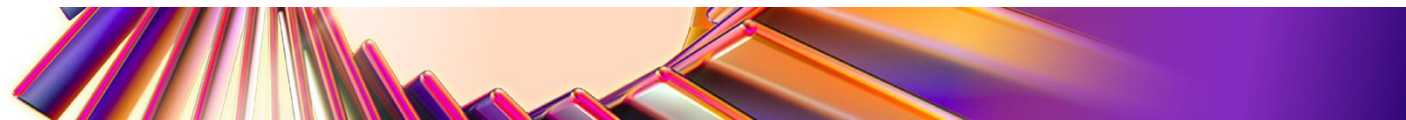
先编写规范，明确需求，再委托 AI 研发实现，结果校验，更少的过程态交互，更长的执行时间，交付更优质的结果

Enhanced Context Engineering

增强对代码工程的理解，更强的检索能力让 AI 更懂复杂工程。通过记忆&个性化上下文，让 AI 在实际编码过程中，更懂得用户行为偏好

DevOps Toolchain Integration

通过 CLI 产品模块，快速与企业内部工具集成对接，成为这些工具的 Agent 核心。复用模型适配、Agent 框架、上下文工程能力，专注企业特有场景实现。



▶ 工程知识不仅仅是代码，有更多的隐性信息没有被记录

过去 Coding Agent 的上下文组成以及面临的问题

手工建立的 Project Rules

编写优秀的 Rules 非常困难

根据用户问题检索的代码

没有架构知识，容易遗漏代码

系统提示词中的编程要求

仅有粗略要求描述

MCP 获取的三方 API 文档

没有企业私有的文档

手工引入的知识库内容

无法及时维护和更新

真实软件开发过程中的期待

自动化提取工程规范

自动化总结工程架构

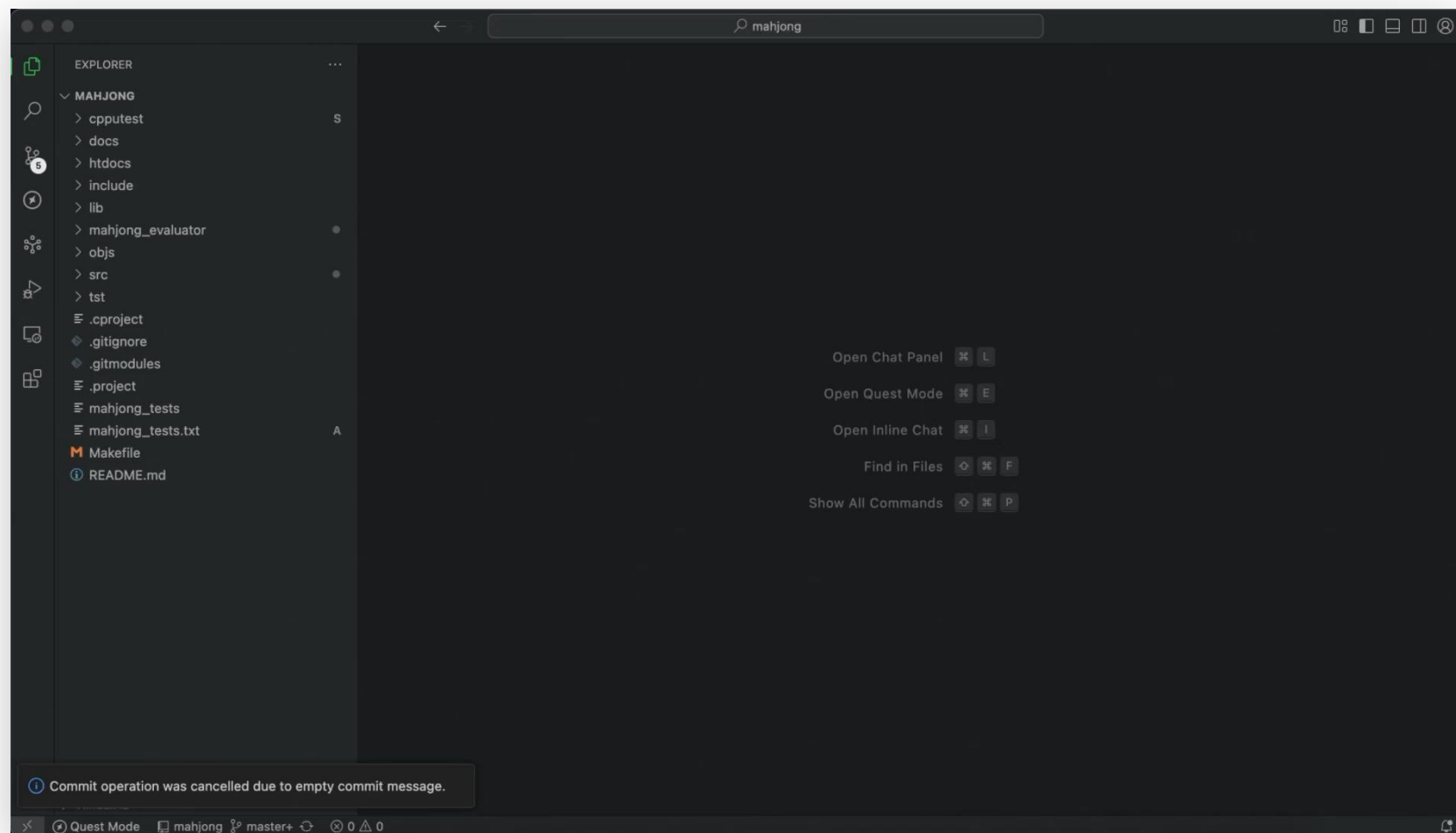
自动学习个人编程偏好

自动化生成私有 API 文档

自动化维护各工程知识库

自动化生成
自动化维护
自动化召回
可审查
可干预
可分享
可复用

▶ AI 驱动的工程知识生成、迭代、运用



从代码到架构的全局理解

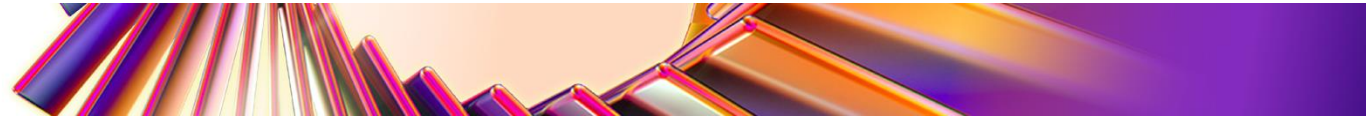
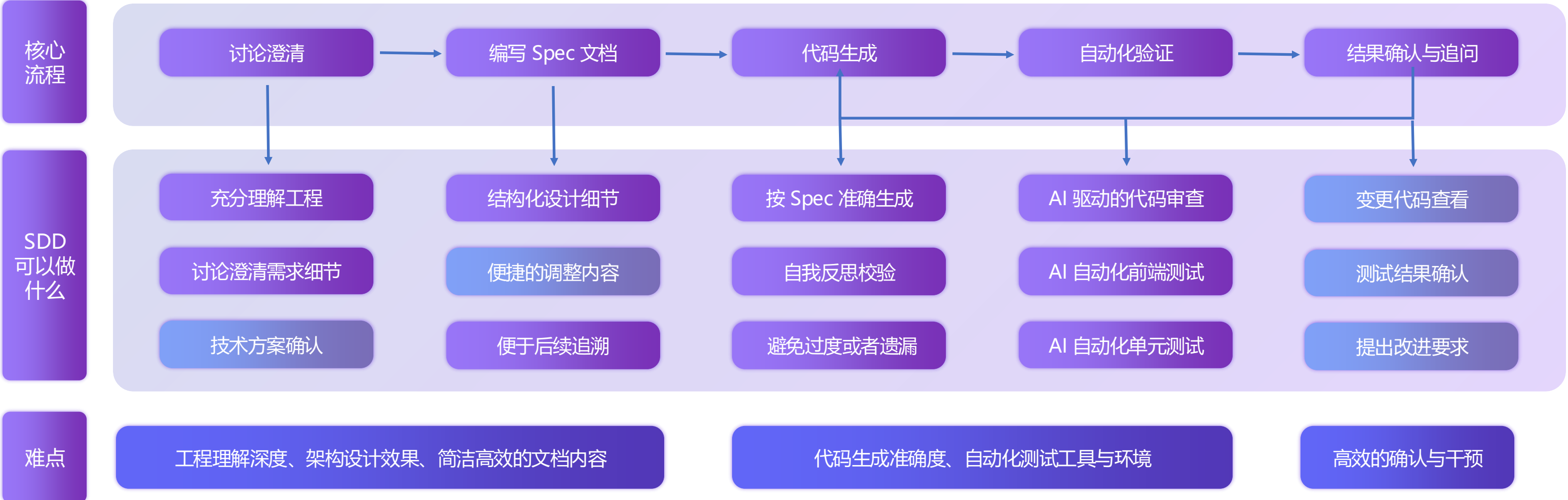
WHY: Agent 仅仅依赖代码检索无法具备对工程或模块的整体架构理解，无法感知隐性的代码规范和设计取舍原则。

HOW: 使用专用模型完成工程架构知识生成和迭代，自动提取开发者偏好和工程规范，指导智能体代码生成。

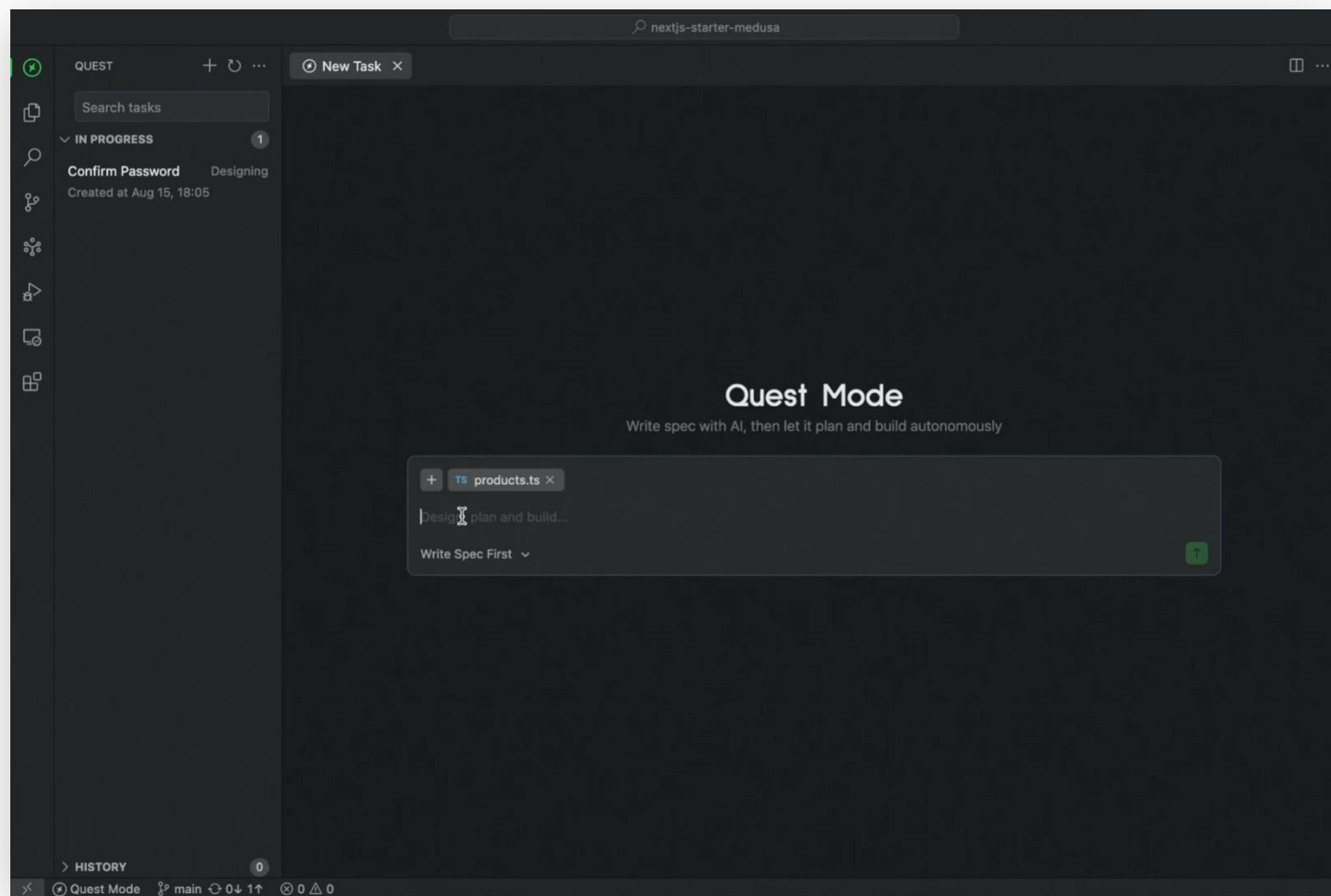
WHAT: Repo wiki、Memory、Code search。



为什么 Spec-Driven Development 会被软件开发者追捧



► SDD 的最佳范式 Quest Mode



设计先行、精准表达

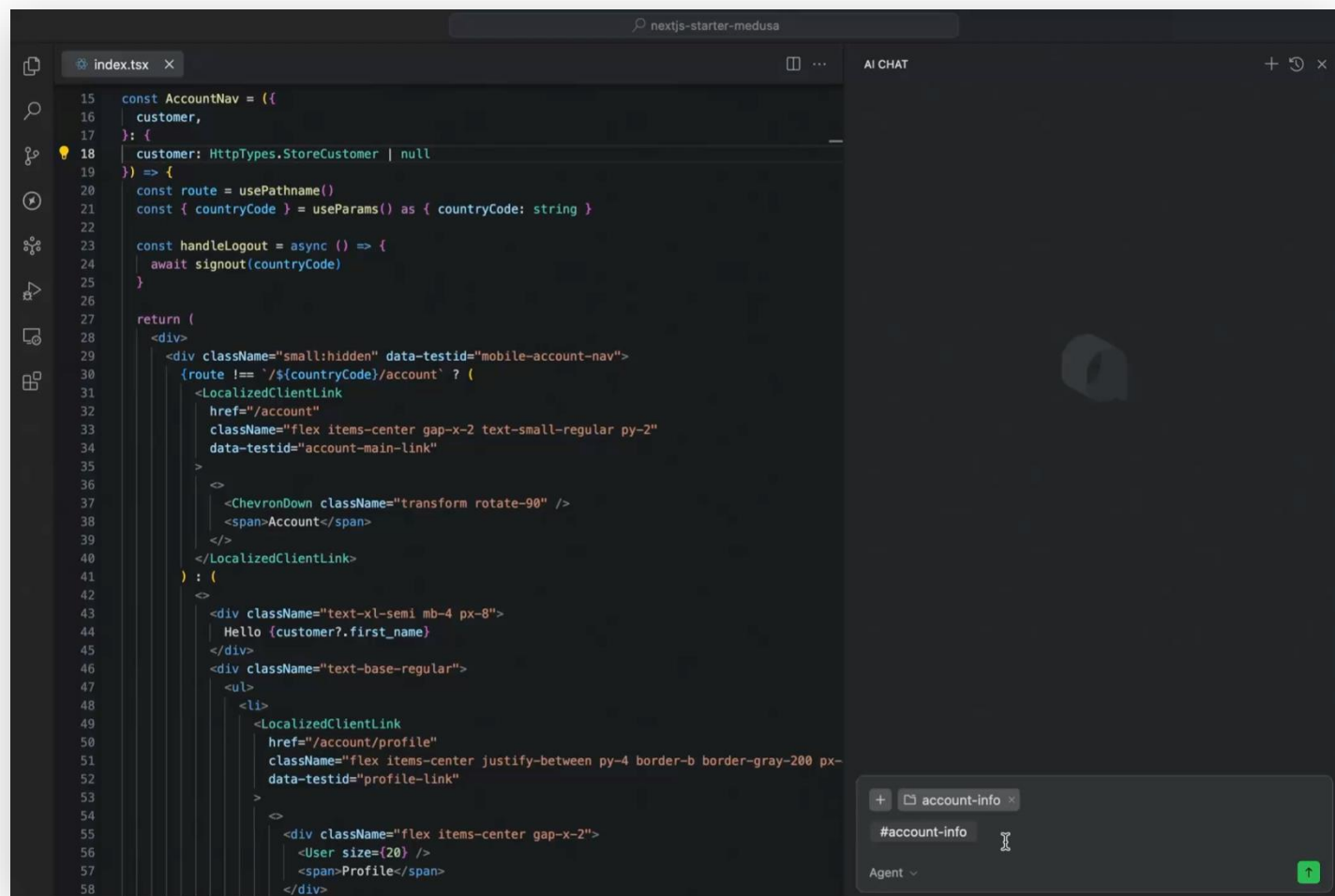
WHY: 没有精准的技术设计约束和前期详尽的工程探索和讨论, 智能体难以高质量交付预期代码。

HOW: 通过 Spec-Driven Development 模式, 实现智能体端到端交付各种编程任务。

WHAT: Quest Mode 配合顶级代码大模型, 多智能体架构, 做到 SDD 模式的生产级落地。



从本地到云端的 Coding Agent



```
15 const AccountNav = ({
16   customer,
17   ...props
18   customer: HttpTypes.StoreCustomer | null
19 }) => {
20   const route = usePathname()
21   const { countryCode } = useParams() as { countryCode: string }
22
23   const handleLogout = async () => {
24     await signout(countryCode)
25   }
26
27   return (
28     <div>
29       <div className="small:hidden" data-testid="mobile-account-nav">
30         {route !== `/${countryCode}/account` ? (
31           <LocalizedClientLink
32             href="/account"
33             className="flex items-center gap-x-2 text-small-regular py-2"
34             data-testid="account-main-link"
35           >
36             <ChevronDown className="transform rotate-90" />
37             <span>Account</span>
38           </LocalizedClientLink>
39         ) : (
40           <div className="text-xl-semi mb-4 px-8">
41             Hello {customer?.first_name}
42           </div>
43           <div className="text-base-regular">
44             <ul>
45               <li>
46                 <LocalizedClientLink
47                   href="/account/profile"
48                   className="flex items-center justify-between py-4 border-b border-gray-200 px-8"
49                   data-testid="profile-link"
50                 >
51                   <div className="flex items-center gap-x-2">
52                     <User size={20} />
53                     <span>Profile</span>
54                   </div>
55                 </LocalizedClientLink>
56               </li>
57             </ul>
58           </div>
59         )
60       </div>
61     </div>
62   )
63 }
```

并行 + 异步带来效率跃升

WHY: 当智能体 + 模型能力越来越强, 时间和空间将成为效率提升的瓶颈, 开发者将逐步脱离智能体 workflow 循环。

HOW: 通过本地 Git Worktree + 安全沙箱、远程标准执行环境, 可实现随时随地委派编码任务, 并行执行。

WHAT: Local Agent + Git Worktree、Remote Agent。

▶ 从代码编写单一场景扩展到 DevOps 全流程

```
~/qoder (0.08s)
ls

~/qoder
qodercli

+ Welcome to Qoder CLI! v0.0.8-preview
cwd: /Users/leon/qoder

Tips for getting started:
1. Ask questions, edit files, or run commands.
2. Be specific for the best results.
3. Type /help for more information.

> 写一个todolist应用
```

将 Qoder Agent 带到每一个终端

WHY: CLI 是 DevOps 的粘合剂，可以让构建、测试、部署等一系列复杂操作被轻松地编排和自动化。

HOW: 通过 CLI、SDK 模式与 DevOps 工具进行集成，复用模型适配、Agent框架、上下文工程能力。

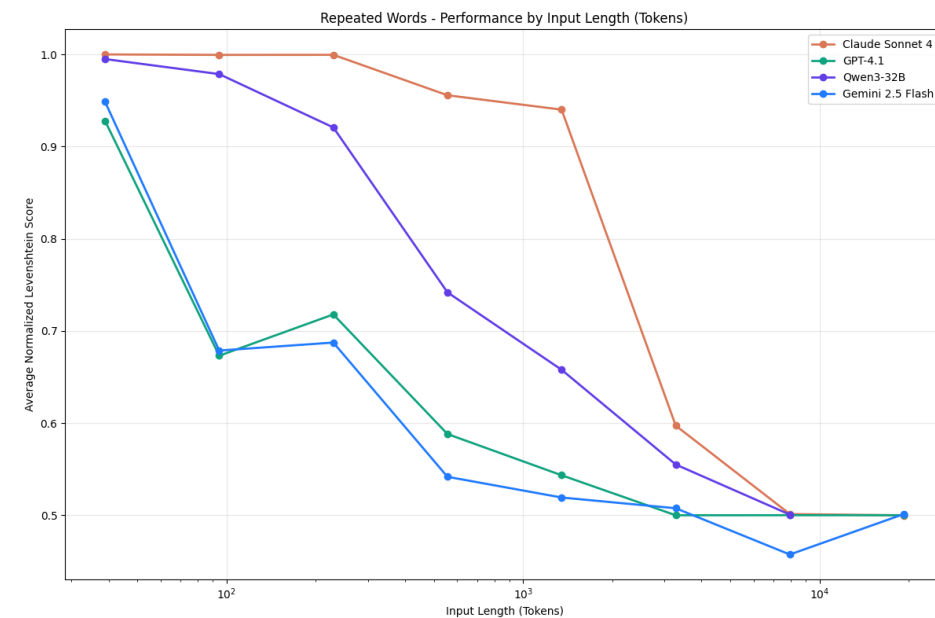
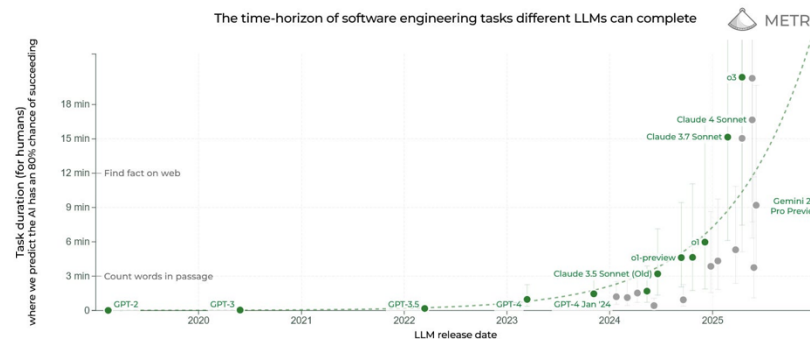
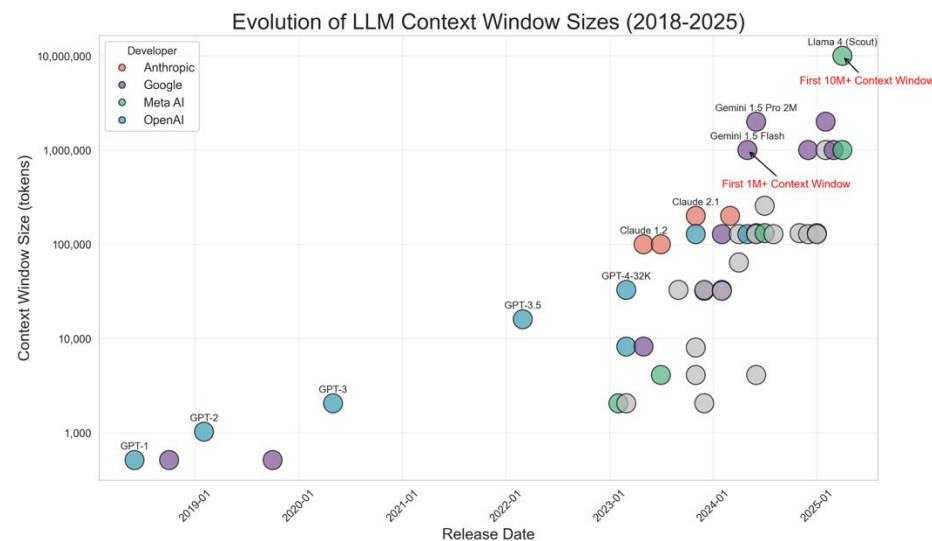
WHAT: CLI 支持 headless 模式可被集成，支持自定义插件、命令及子代理的扩展。



PART 03

Qoder 上下文工程技术优化实践

上下文优化的目标：有限上下文空间、最大信息密度、最有效结构组织

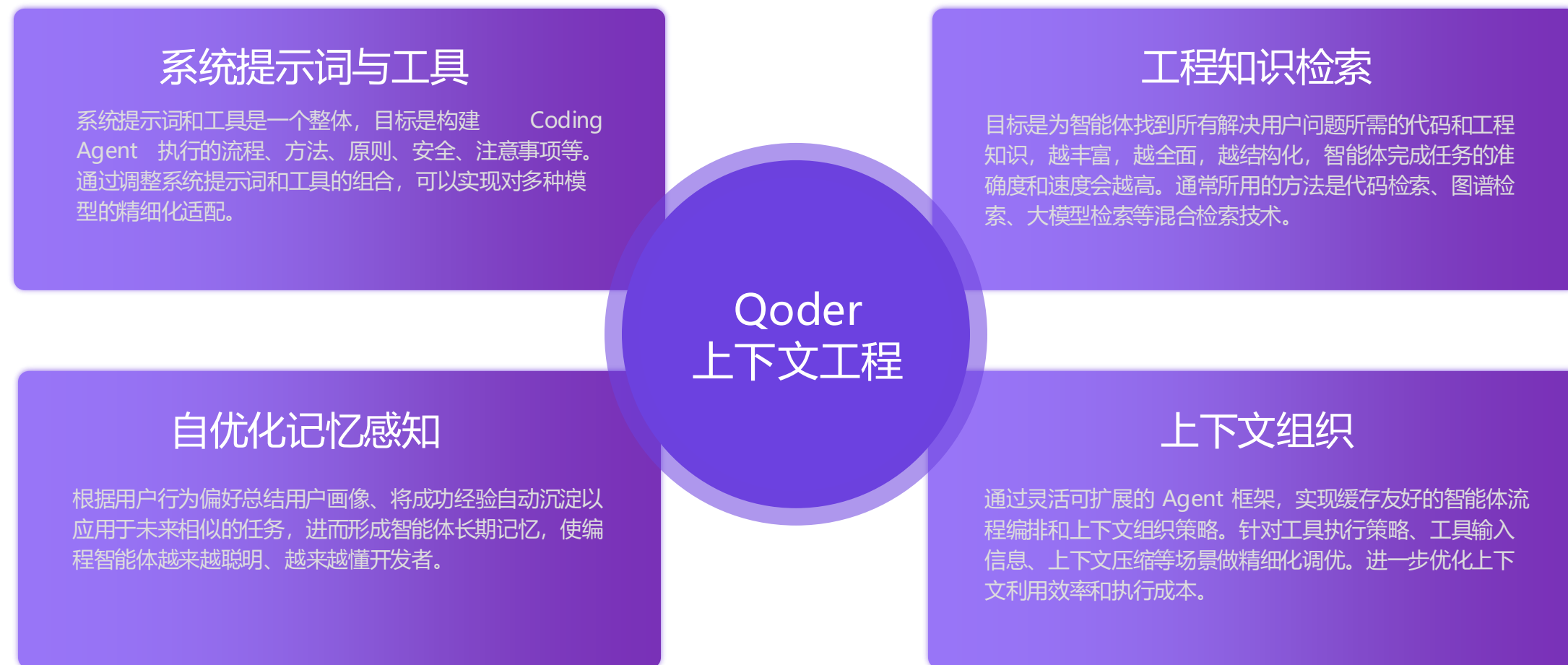


各模型厂商支持的上下文长度都在快速扩大

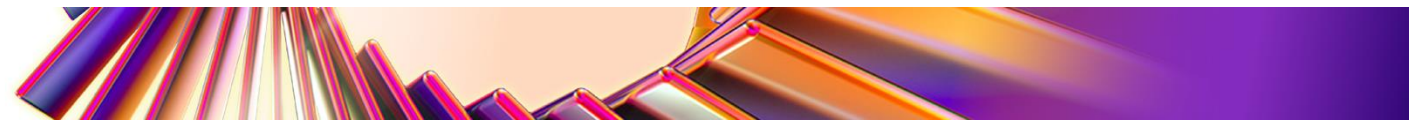
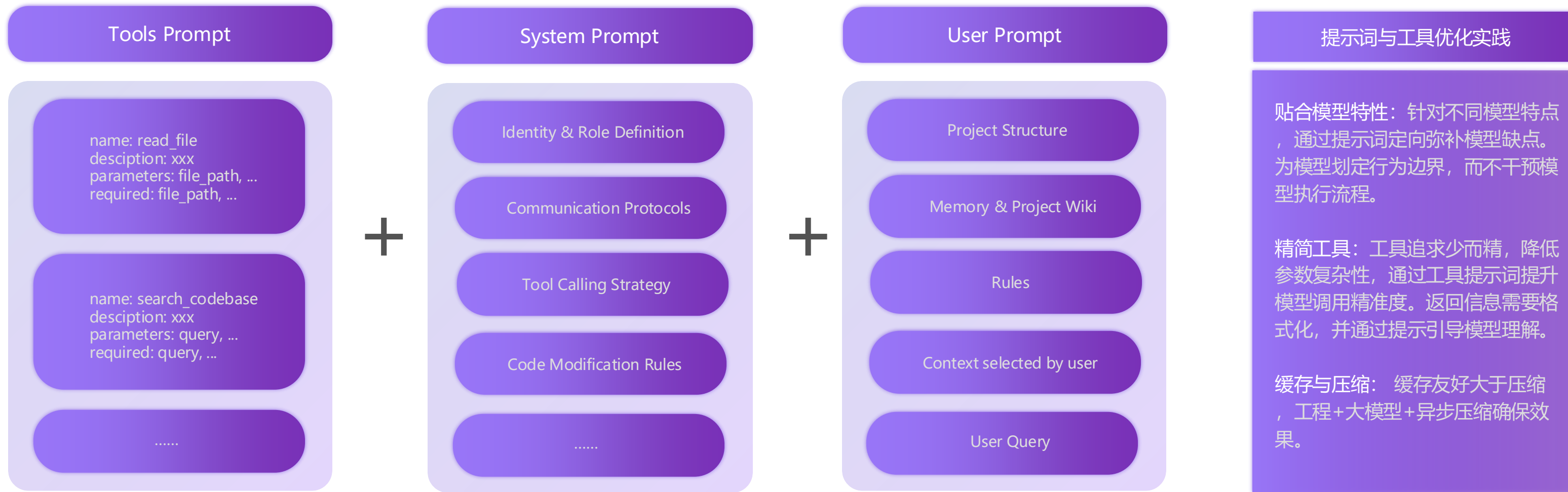
模型可以完成的更长时间的编码任务

模型性能随着上下文长度增长却呈现快速下降趋势

<https://research.trychroma.com/context-rot>
<https://www.oneusefulthing.org/p/the-bitter-lesson-versus-the-garbage>
<https://www.meibel.ai/post/understanding-the-impact-of-increasing-llm-context-windows>



► 精细的系统提示词与工具调优



代码与架构知识融合检索引擎



工程知识检索优化实践

代码语义检索：语义检索仍然必要，在大工程中配合代码图谱相比文本+大模型检索效果更优。

工程知识检索：代码检索融合架构知识检索可以为大模型补充高层次知识，进一步规范模型生成结果，提升代码架构一致性。

Agentic Search：大模型驱动的检索引擎相比工作流效果更好，但需要突破检索速度瓶颈。

► 具备自优化能力的记忆感知技术

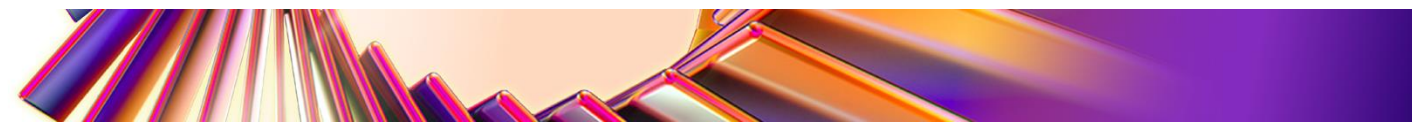


记忆感知难点与探索

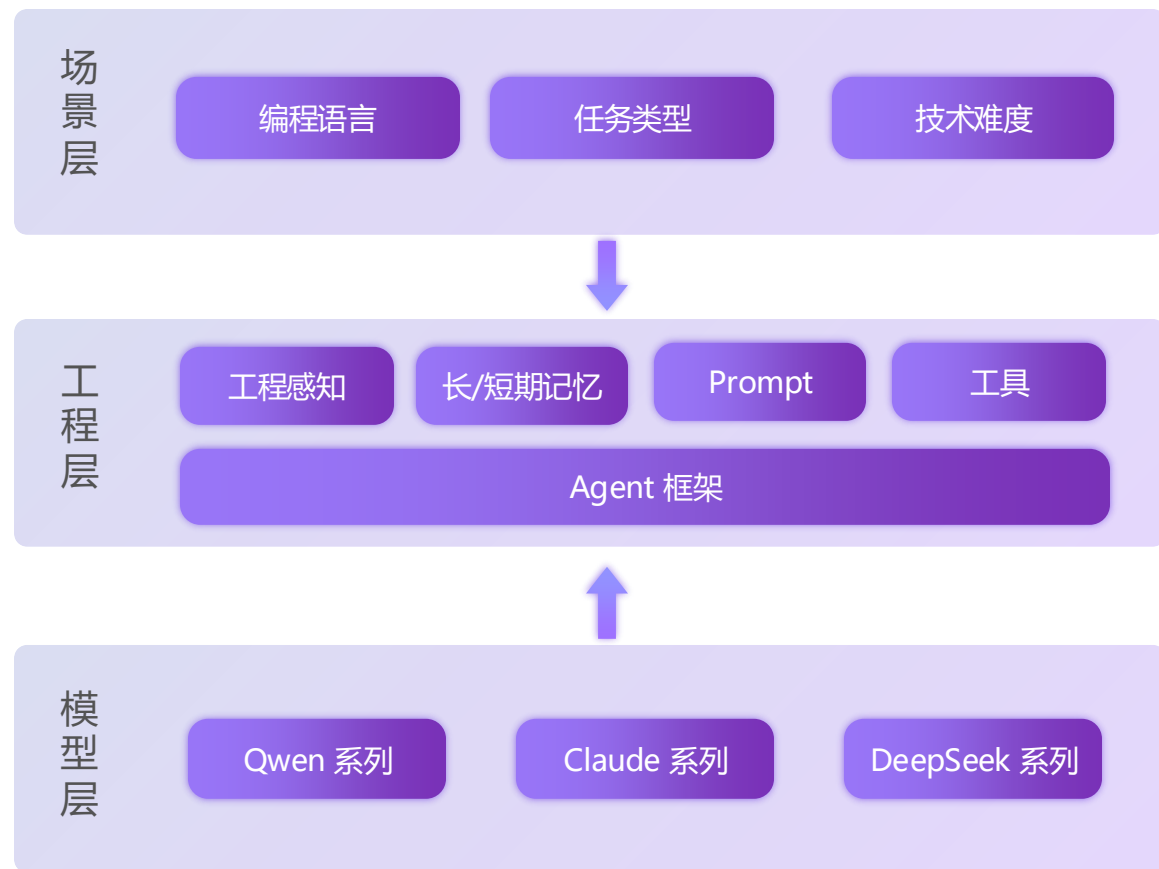
难点: 面临记忆提取检索准确性不足和记忆整理效率低下的双重挑战，难以在海量历史信息中定位关键内容，导致上下文理解偏差。

探索: 多阶段记忆提取与召回机制。让智能体自主决策何时召回，有效避免冗余信息干扰。

探索: 对存储记忆进行质量评估和有效性分析，依据评价指标智能遗忘低价值信息并融合重复记忆。



▶ 自动化评测驱动效果持续改进

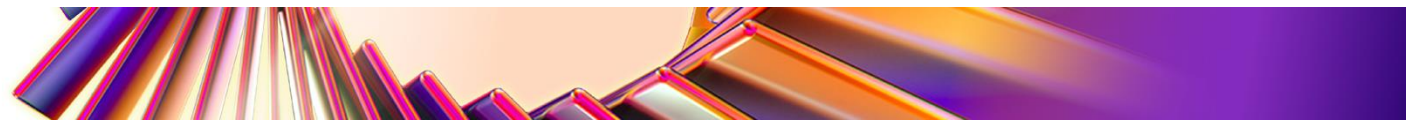


评测是一切优化的核心

测试集选择：尽量拟合线上语言分布、任务类型分布、仓库难度分布，构建真实软件测试集。

评测维度：需要覆盖生成准确率、代码规范性、测试完整性、运行时间、迭代轮数、token消耗等多维度。

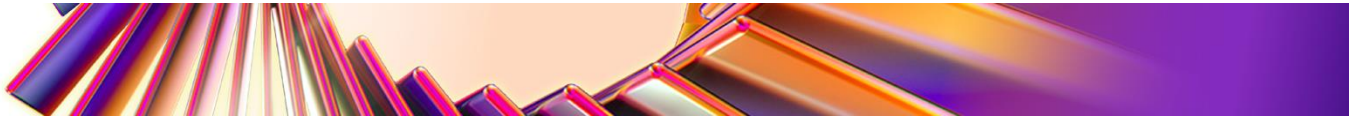
在线评测：重点关注代码保留率、异常率、对话成本、用户满意度、请求时间、调用工具次数等维度。



► Qoder Coding Agent 整体架构和优化方向



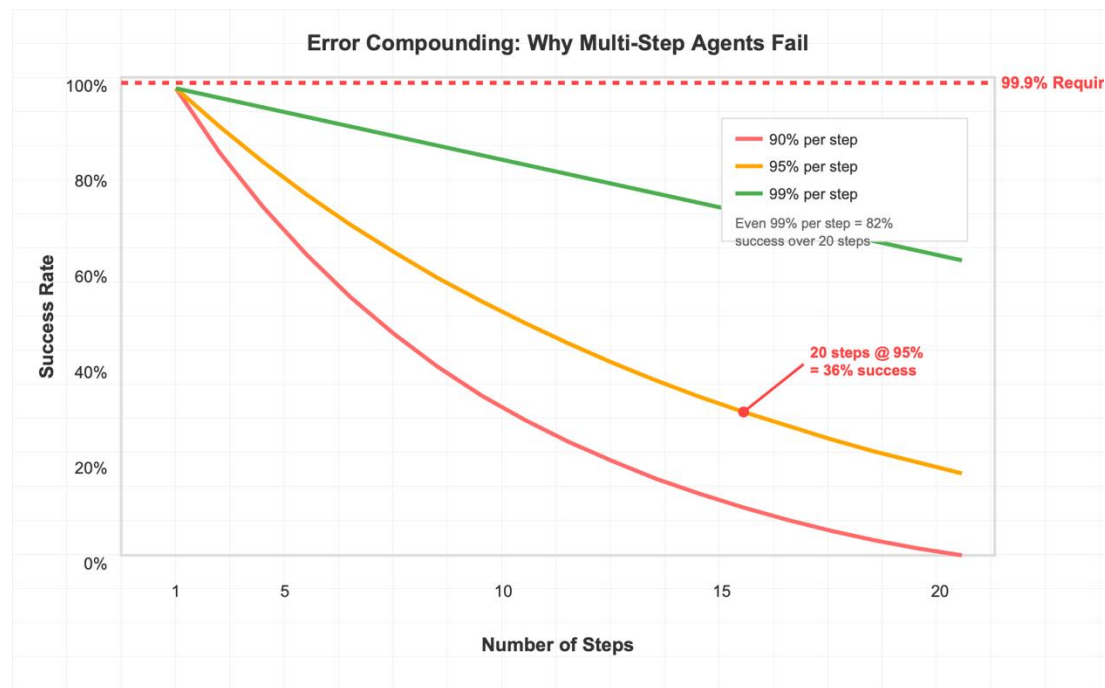
- 整体设计和优化目标
- 生产级代码生成能力
 - 更少的开发者参与
 - 更高的并行化执行
 - 更全面的开发流程覆盖



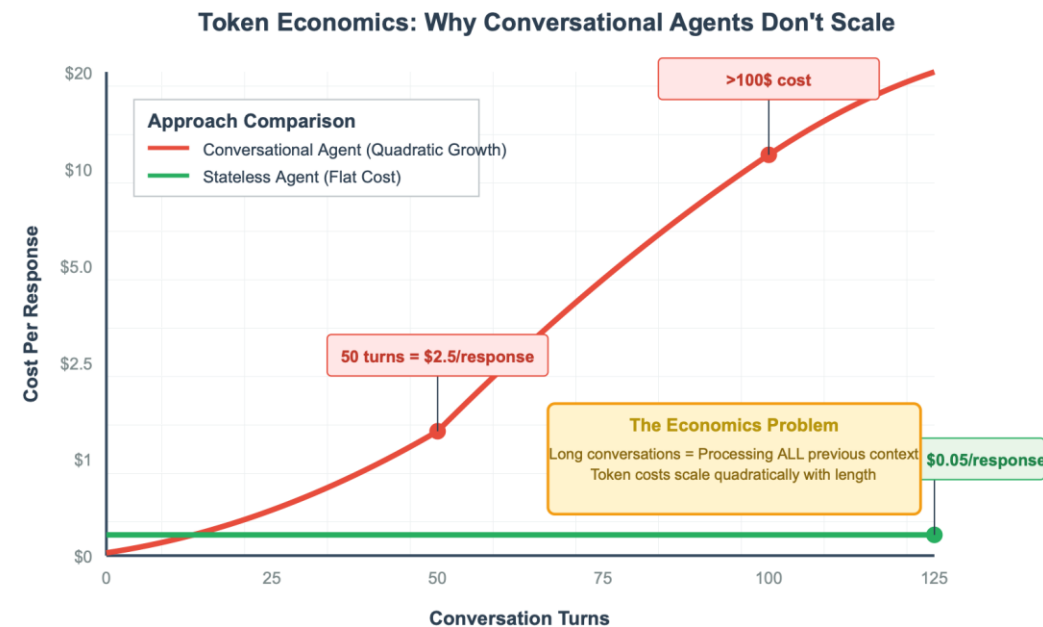
PART 04

未来技术挑战与应对

长任务代码生成准确率、成本的挑战与应对

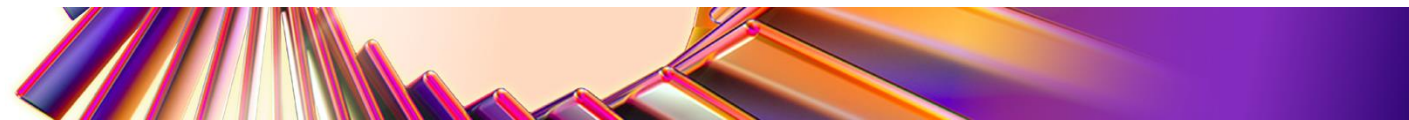


通过强化反思、迭代、验证机制去解决智能体准确率问题

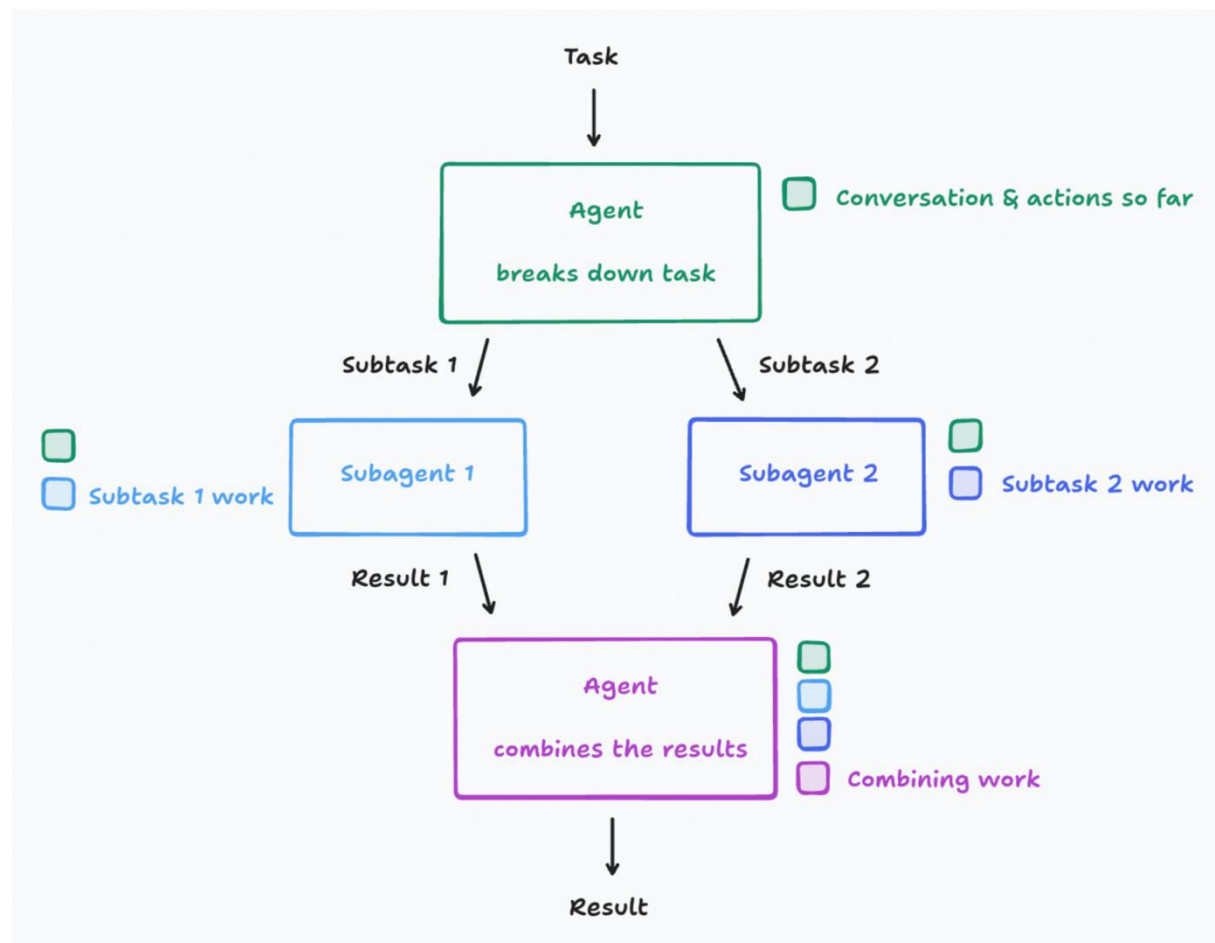


通过强化工具、上下文管理、多模型调度去解决成本问题

图表引用: <https://utkarshkanwat.com/writing/betting-against-agents>



多智能体架构在复杂任务中的挑战与应对



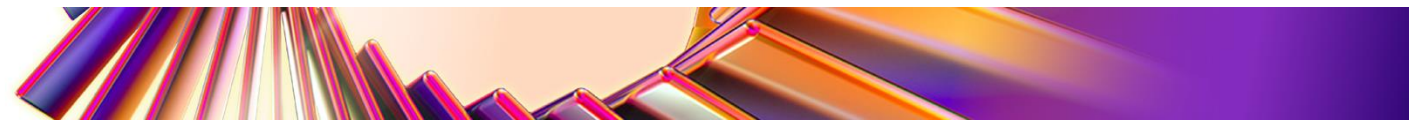
问题、挑战、应对

关键问题：在 Coding 任务中是否需要引入多智能体架构，其收益和问题有哪些，适用于什么场景？

关键挑战：多智能体架构可以有效解决单智能体上下文窗口受限问题，但引入了多智能体之间信息共享、同步问题，反而让多智能体架构更易失败，消耗更多token，速度更慢。

应对思路：独立边界任务，完整的信息传递，专用模型，谨慎采用并行化。

图表引用: <https://cognition.ai/blog/dont-build-multi-agents#a-theory-of-building-long-running-agents>



未来上下文能力的突破在于与模型能力的一体化



科技生态圈峰会 + 深度研习 —— 1000+ 技术团队的选择



NiDD 峰会



NDD AI+研发数字峰会
上海站

时间 | 2026.05.22-23



NDD AI+研发数字峰会
北京站

时间 | 2026.08.21-22



NDD AI+研发数字峰会
深圳站

时间 | 2026.11.20-21



AiDD峰会详情

NiDD x AI+产品峰会

时间 | 2026.01.16-17
NiDD x AI+产品创新峰会

上海站



时间 | 2026.08.23-24
NiDD x AI+产品创新峰会

北京站



产品峰会详情



AI+ PRODUCT INNOVATION SUMMIT 01.16-17 · ShangHai

2026 AI+产品创新峰会



扫码查看会议详情



Track 1: AI 产品战略与创新设计

从0到1的AI原生产品构建

论坛1: AI时代的用户洞察与需求发现

论坛2: AI原生产品战略与商业模式重构

论坛3: Agentic AI产品创新与交互设计



Track 2: AI 产品开发与工程实践

从1到10的工程化落地实践

论坛1: 面向Agent智能体的产品开发

论坛2: 具身智能与AI硬件产品

论坛3: AI产品出海与本地化开发



Track 3: AI 产品运营与智能演化

从10到100的AI产品运营

论坛1: AI赋能产品运营与增长黑客

论坛2: AI产品的数据飞轮与智能演化

论坛3: 行业爆款AI产品案例拆解

2-hour Speech: 回归本质



用户洞察的第一性

——2小时思维与方法论工作坊

在数字爆炸、AI快速发展的时代，
仍然考验“看见”的“同理心”

Panel 1: 出海前瞻



“出海避坑地图”圆桌对话

——不止于翻译：

AI时代的出海新范式

Panel 2: 失败复盘



为什么很多AI产品“叫好不叫座”？

——从伪需求到真价值：

AI产品商业化落地的关键挑战

智能重构产品 数据驱动增长
Reinventing Products with Intelligence, Driven by Data

80+
业界大咖
汇聚产品大咖的真知灼见

40+
创新案例
开拓全新AI+产品视角

10+
分论坛
涵盖产品到商业增长的全链路

800+
听众规模
共同探索产品的智能重构



第8届 AI+研发数字峰会

AI+ Development Digital Summit



感谢聆听！

扫码领取会议PPT资料

